**Heuristic Analysis**

Between the chosen breadth-first, depth-first, and uniform-cost search methods, depth-first search consistently performed the worst. Depth-first search performs suboptimally because is traverses a tree vertically and can't choose to follow nodes that are favorable since search it is not done horizontally as in breadth-first search. Of course, depth-first search finds a solution significantly faster than breadth-first, such as in *Problem 1* (0.03 vs. 0.061 seconds), and this is one of it's benefits. Nonetheless, the solutions it finds are not optimal because of the comparably larger plan lengths. On the other hand, breadth-first search performs better because the shortest path is always considered first[1].

Of the heuristic searches, A*'s level_sum consistently performed the worst; it took the longest to evaluate and resulted in plan lengths equal to much more time-efficient search methods. This is likely because of its high computational complexity, relative to the other search methods. To the contrary, ignore_preconditions performed the best for all but *Problem 1*.

Between problems, there wasn't any consistency between heuristic and non-heuristic models. In other words, heuristic searches performed just slightly better in *Problem 1*, while in *Problem 3,* the opposite is true. One pattern that may be emerging is as a problem requires more time to complete, heuristic approach perform increasingly better. Moreover, the average amount of time taken to complete increases from *Problem 1* to *Problem 2* to *Problem 3*. This would make sense. Heuristic approaches add an extra layer of complexity that are time consuming to evaluate. And in cases where the problems are relatively simple, that added complexity means by the time the heuristic portion of the model has been evaluated, non-heuristic models are already finished.

From the results below, breadth-first search is optimal for very simple problems and even to most heuristics. The exception is A* ignore_preconditions.

[1] RUSSELL, STUART NORVIG PETER. ARTIFICIAL INTELLIGENCE: a Modern Approach. PEARSON, 2018.

| *Problem 1 Results* | | | | | | |
|---|---|---|---|---|---|---|
| | **Optimal?** | **Expansions** | **Goal Tests** | **New Nodes** | **Plan Length** | **Time Elapsed (s)** |
| **Breadth First Search** | Yes | 43 | 56 | 180 | 6 | 0.061589 |
| **Depth First Graph Search** | No | 21 | 22 | 84 | 20 | 0.0255058 |
| **Uniform Cost Search** | Yes | 55 | 57 | 227 | 6 | 0.063651 |
| **A* Search h_1** | Yes | 55 | 57 | 224 | 6 | 0.07220 |
| **A* Search h_ignore_preconditions** | Yes | 41 | 43 | 170 | 6 | 0.071130 |
| **A* Search h_pg_levelsum** | Yes | 11 | 13 | 50 | 6 | 1.4855884 |
| *Problem 2 Results* | | | | | | |
| | **Optimal?** | **Expansions** | **Goal Tests** | **New Nodes** | **Plan Length** | **Time Elapsed (s)** |
| **Breadth First Search** | Yes | 3343 | 4609 | 30509 | 9 | 12.84688 |
| **Depth First Graph Search** | No | 436 | 437 | 3862 | 434 | 3.2067 |
| **Uniform Cost Search** | Yes | 4852 | 4854 | 44030 | 9 | 18.15301 |
| **A* Search h_1** | Yes | 4852 | 4854 | 44030 | 9 | 21.33570 |
| **A* Search h_ignore_preconditions** | Yes | 1450 | 1452 | 13303 | 9 | 7.8936686 |
| **A* Search h_pg_levelsum** | Yes | 86 | 88 | 841 | 9 | 253.455475 |
| *Problem 3 Results* | | | | | | |
| | **Optimal?** | **Expansions** | **Goal Tests** | **New Nodes** | **Plan Length** | **Time Elapsed (s)** |
| **Breadth First Search** | Yes | 14663 | 18098 | 129631 | 12 | 70.12465 |
| **Depth First Graph Search** | No | 3767 | 3768 | 31703 | 3587 | 102.905 |
| **Uniform Cost Search** | Yes | 18223 | 18225 | 159618 | 12 | 83.244159 |
| **A* Search h_1** | Yes | 18223 | 18225 | 159618 | 12 | 92.7839 |
| **A* Search h_ignore_preconditions** | Yes | 5040 | 5042 | 44944 | 12 | 29.5823 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **A* Search h_pg_levelsum** | Yes | 325 | 327 | 3002 | 12 | 1304.85175 |

*The most efficient search type is highlighted in gray.*

| *Optimal Plan* | | |
|---|---|---|
| | **Search Type** | **Action Sequence** |
| **Problem 1** | Breadth First Search | Load(C1, P1, SFO)<br>Load(C2, P2, JFK)<br>Fly(P2, JFK, SFO)<br>Unload(C2, P2, SFO)<br>Fly(P1, SFO, JFK)<br>Unload(C1, P1, JFK) |
| **Problem 2** | A* Search h_ignore_preconditions | Load(C3, P3, ATL)<br>Fly(P3, ATL, SFO)<br>Unload(C3, P3, SFO)<br>Load(C1, P1, SFO)<br>Fly(P1, SFO, JFK)<br>Unload(C1, P1, JFK)<br>Load(C2, P2, JFK)<br>Fly(P2, JFK, SFO)<br>Unload(C2, P2, SFO) |
| **Problem 3** | A* Search h_ignore_preconditions | Load(C2, P2, JFK)<br>Fly(P2, JFK, ORD)<br>Load(C4, P2, ORD)<br>Fly(P2, ORD, SFO)<br>Unload(C4, P2, SFO)<br>Load(C1, P1, SFO)<br>Fly(P1, SFO, ATL)<br>Load(C3, P1, ATL)<br>Fly(P1, ATL, JFK)<br>Unload(C3, P1, JFK)<br>Unload(C2, P2, SFO)<br>Unload(C1, P1, JFK) |